

PHP

# why PHP?

- free
- web application
- popular

安裝 PHP

# LAMP

MAMP & MAMP PRO - Downlo...

MAMP

localhost:8888 / localhost | ph

localhost:8888

W LAMP (software bundle) - Wiki

Secure

https://en.wikipedia.org/wiki/LAMP\_(software\_bundle)

Not logged in

Talk

Contributions

Create account

Log in

Article

Talk

Read

Edit

View history

Search Wikipedia

WIKIPEDIA

The Free Encyclopedia

Main page

Contents

Featured content

Current events

Random article

Donate to Wikipedia

Wikipedia store

Interaction

Help

About Wikipedia

Community portal

Recent changes

Contact page

Tools

What links here

Related changes

Upload file

Special pages

Permanent link

Page information

Wikidata item

Cite this page

LAMP (software bundle)

From Wikipedia, the free encyclopedia

**LAMP** is an archetypal model of web service [stacks](#), named as an [acronym](#) of the names of its original four [open-source](#) components: the [Linux operating system](#), the [Apache HTTP Server](#), the [MySQL relational database management system](#) (RDBMS), and the [PHP programming language](#). The LAMP components are largely interchangeable and not limited to the original selection. As a solution stack, LAMP is suitable for building [dynamic web sites](#) and [web applications](#).<sup>[1]</sup>

Since its creation, the LAMP model has been adapted to other componentry, though typically consisting of [free and open-source software](#). For example, an equivalent installation on the [Microsoft Windows](#) family of operating systems is known as [WAMP](#) and an equivalent installation on [macOS](#) is known as [MAMP](#).

Contents [hide]

1 Overview

1.1 Variants

2 Software components

2.1 Linux

2.2 Apache

2.3 MySQL and alternatives

2.4 PHP and alternatives

3 High availability and load balancing

4 See also

5 References

6 External links

Web cache

Squid

Polipo

Traffic server

Web server

Apache

Cherokee

Lighttpd

Nginx

CGI scripting

Perl

PHP

Python

Database

MariaDB

MySQL

Drizzle

Linux kernel

AppArmor

SELinux

Smack

TOMOYO

Process Scheduler

Netfilter

Linux network stack

Network scheduler

NIC device driver

kmod-fs-ext4

kmod-fs-btrfs

Lustre

...

Hardware

CPU & RAM

Networking hardware

Storage

SATA

SAS

RAID

iSCSI

NAS

Environment: CCC

Crackers

Attacks

Requests

Responses

Internet

Customers

Competitors

Botnets

A high-level overview of LAMP's building blocks and overall system environment, displayed here in combination with optionally used web caches.



# MAMP

MAMP & MAMP PRO - Downlo x

MAMP

localhost:8888 / localhost | ph x

localhost:8888

LAMP (software bundle) - Wiki x

TAKER

localhost:8888/MAMP/?language=English

Start

My Website


phpInfo

Tools ▾

FAQ


MAMP Website

Buy MAMP PRO



## Welcome!

MAMP has been installed successfully.



### PHP

[phpinfo](#) shows the current configuration of PHP.

### MySQL

MySQL can be administered with [phpMyAdmin](#).

To connect to the MySQL server from your own scripts use the following connection parameters:

Host	localhost
Port	3306

### MAMP Version

4.2 → Your version is up-to-date.

### News

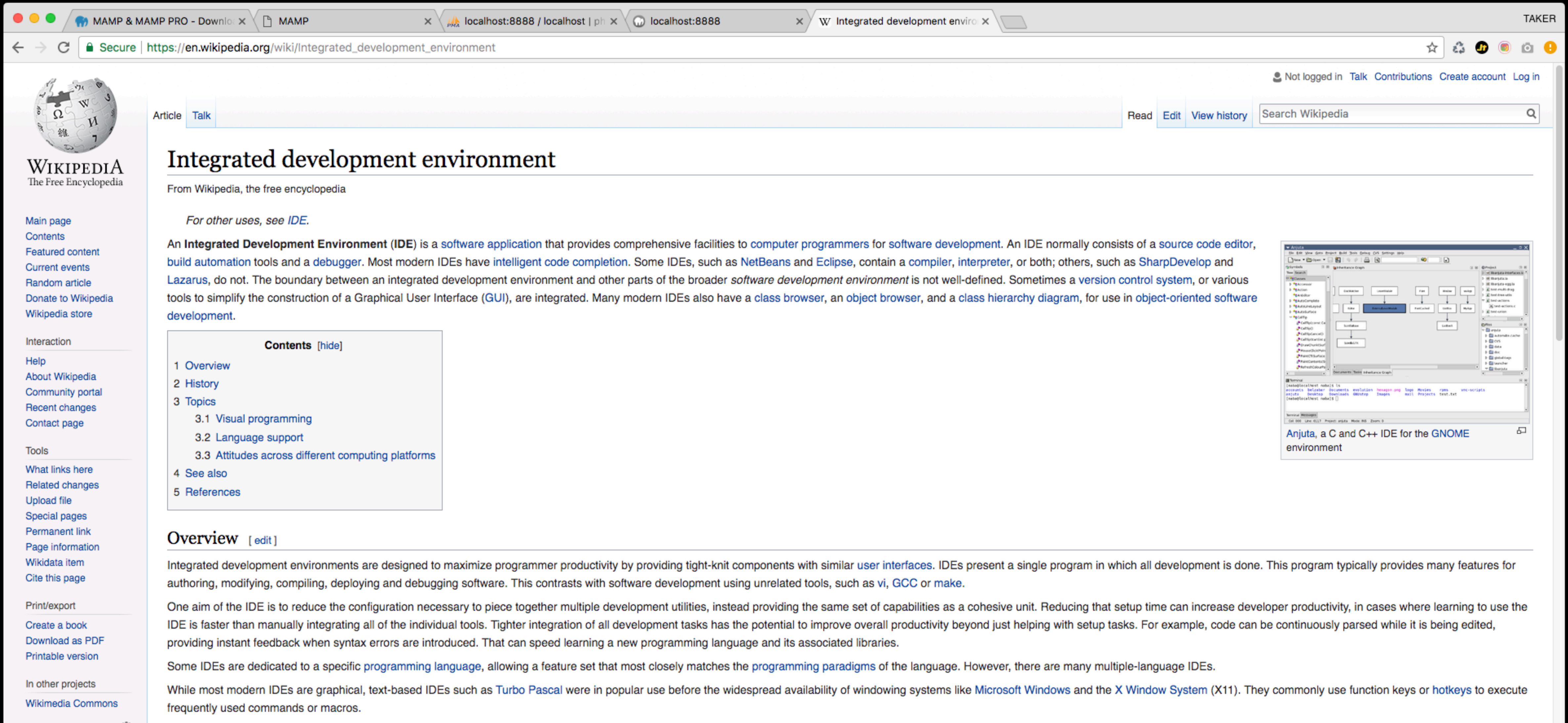
No news available at the moment!

# 安裝 IDE

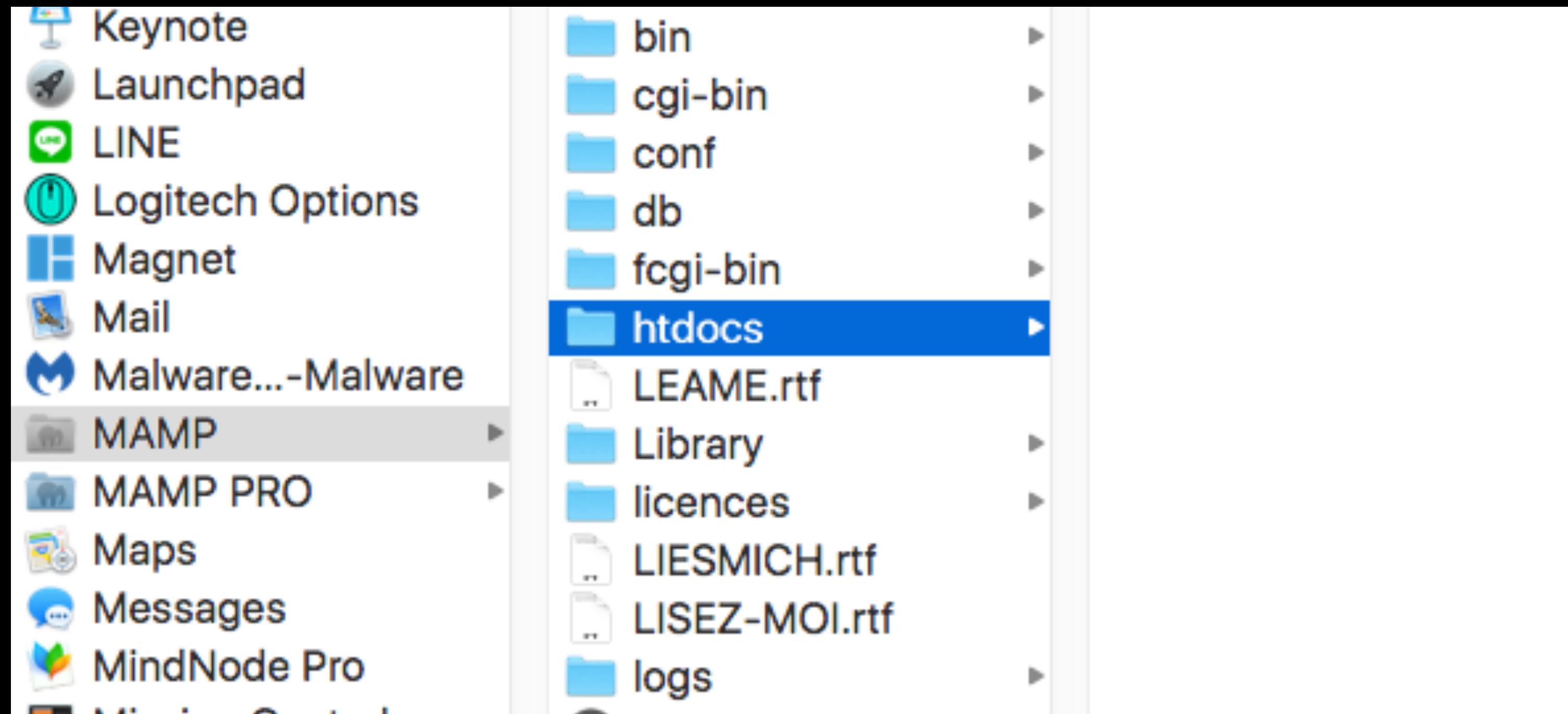
- VS code
- Sublime Text 3
- Textmate



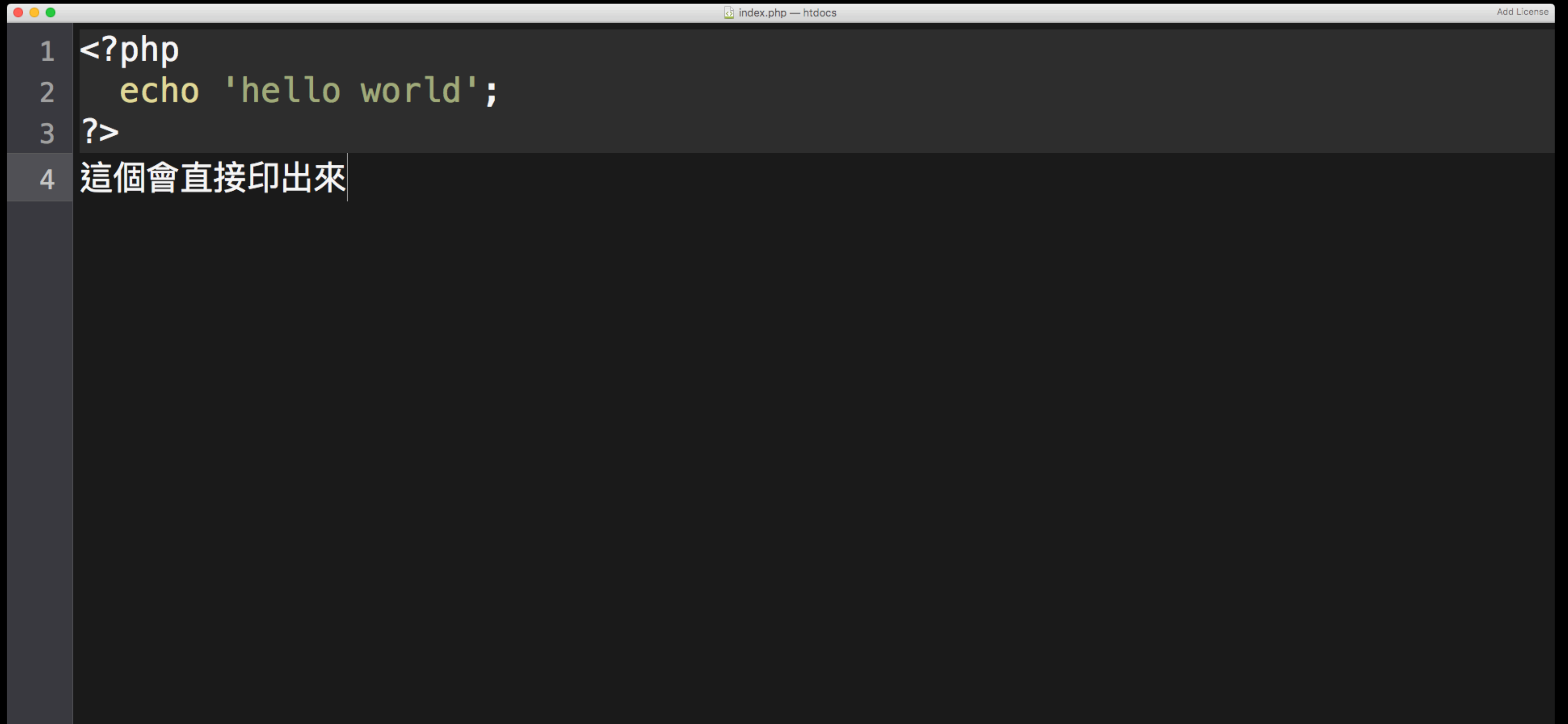
# 安裝 IDE



# hello world



# hello world



A screenshot of a code editor window. The title bar at the top shows three colored window control buttons (red, yellow, green) on the left, the text "index.php — htdocs" in the center, and "Add License" on the right. The editor area has a dark background with light gray line numbers on the left. The code is as follows:

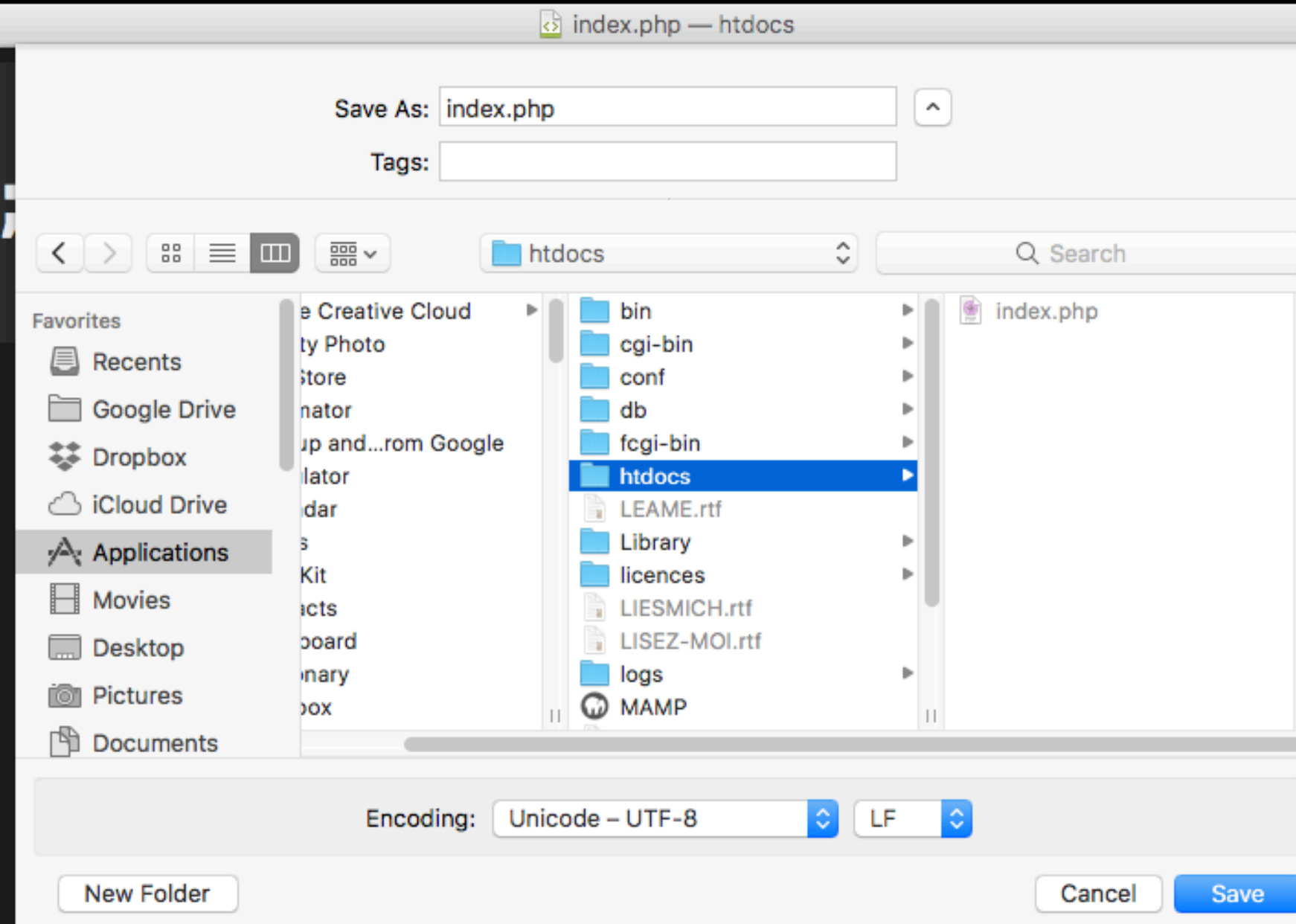
```
1 <?php
2     echo 'hello world';
3 ?>
4 這個會直接印出來
```



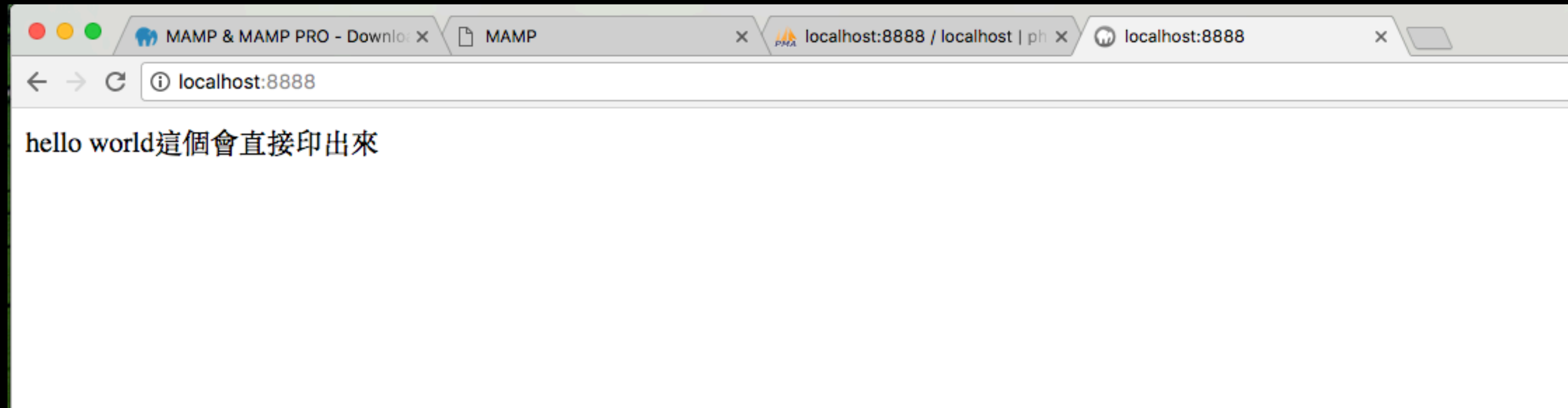
# hello world

```
1 <?php
2     echo 'hello world';
3 ?>
```

4 這個會直接印出來



# hello world





**Mix with HTML**

# Mix with HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>

  hello world

</body>
</html>
```

# Mix with HTML

```
<?php  
echo 'hello world';  
?>
```

# Mix with HTML

```
<?php  
    echo 'hello world';  
?>
```

這個會直接印出來

# Mix with HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>

  <?php echo 'hello world'; ?>

</body>
</html>
```

# Mix with HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>

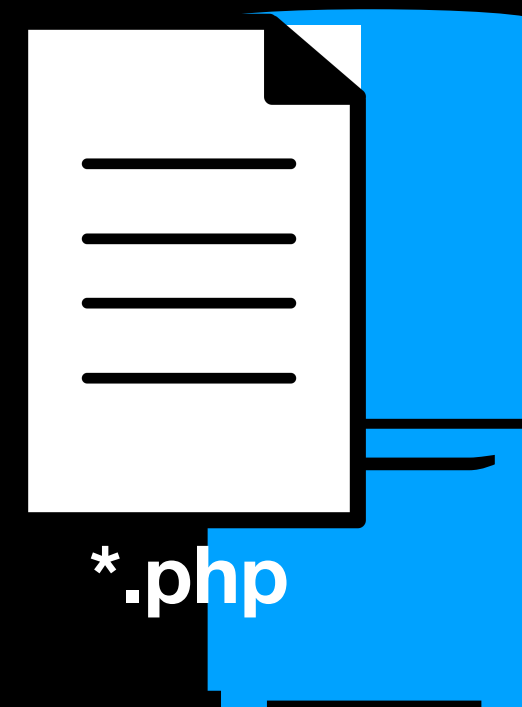
  hello world

</body>
</html>
```

# Mix with HTML



HTML/CSS/JS



\*.php



# Mix with HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>

  <?php echo 'hello world'; ?>

</body>
</html>
```

# Mix with HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>

  <?php echo 'hello world'; ?>

</body>
</html>
```

# Mix with HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>

  hello world

</body>
</html>
```

# Mix with HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>

  hello world

</body>
</html>
```

# Mix with HTML



HTML/CSS/JS



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>

  hello world

</body>
</html>
```

# Mix with HTML



# Mix with HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>

  <?php echo 'hello world'; ?>

</body>
</html>
```



# Mix with HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  ...
</body>
<script>
  var foo = 'hello world';
</script>
</html>
```

# Mix with HTML



# Mix with HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  ...
</body>
<script>
  var foo = '<?php echo 'hello world' ?>';
</script>
</html>
```

# Mix with HTML



Basic

# PHP start & end tags

```
<?php
```

```
    echo "Hello, this is PHP!";
```

```
?>
```

# data

variable, string & numbers



# Variable

```
$foo = 'this is a string';  
$bar = "this is also a string";
```

```
$number = 3.579;  
$negative = -99987;
```

```
$boolean = true;  
$booleanFalse = false;
```

# text

```
$foo = ', yeah!';
```

```
echo 'this is a string';
```

```
echo "this is also a string$foo";
```

```
// this is also a string, yeah!
```

# text

```
echo 'This is he\'s book';  
echo "Love Quote of the Day: \"To love is to act.\"";  
  
echo 'This is he's book';  
echo "Love Quote of the Day: \"To love is to act.\"";
```

# number

```
echo 123;  
echo -32768;  
echo 99.87;  
echo 0;
```

# number

```
echo 35 + 22;    // 57
echo 35 - 22;    // 13
echo 22 - 35;    // -13
echo 5 * 6;      // 30
echo 6 / 2;      // 3

echo 7 % 3;      // 1
```

# logic

making decisions & repeating yourself

# true & false

```
// true  
true;  
25;  
99.7;  
-33;  
'this is also true';
```

```
// false  
false;  
null;  
0;  
0.0;  
'';  
'0';
```

# making decisions

```
if ($logged_in) {  
    echo 'welcome, my friend.';  
}  
else {  
    echo 'please login, stranger.';  
}
```



# making decisions

```
$time = 20;  
if ($time > 18) {  
    echo 'turn on the light.';  
}  
else {  
    echo 'turn off the light.';  
}
```

# making decisions

```
echo 1 == 1;  
echo 1 != 1;  
echo 2 > 1;  
echo 2 < 1;  
echo 2 >= 2;  
echo 2 <= 2;
```

# repeat yourself

```
<select>  
<option>1</option>  
<option>2</option>  
<option>3</option>  
<option>4</option>  
<option>5</option>  
<option>6</option>  
<option>7</option>  
<option>8</option>  
<option>9</option>  
<option>10</option>  
</select>
```

# repeat yourself

```
$i = 1;

echo "<select>\n";
while ($i <= 10) {
    echo "<option>$i</option>\n";
    $i++;
}
echo '</select>';
```

# repeat yourself

```
$i = 1;
```

```
echo "<select>\n";
```

```
while ($i <= 10) {
```

```
    echo "<option>$i</option>\n";
```

```
    $i++;
```

```
}
```

```
echo '</select>';
```

# repeat yourself

```
echo "<select>\n";  
for ($i = 1; $i <= 10; $i++) {  
    echo "<option>$i</option>\n";  
}  
echo '</select>';
```

# array

groups of data

# array

key	value
apple	red
lemon	green
pinapple	yellow



# array

key	value
0	sunny
1	cloudy
2	raining

# dictionary

key	value
apple	red
lemon	green
pineapple	yellow

# array

key	value
apple	red
lemon	green
pineapple	yellow

key	value
0	sunny
1	cloudy
2	raining

# array

```
$fruits = array('apple' => 'red',  
               'lemon' => 'green',  
               'pineapple' => 'yellow');  
echo $fruits['apple']; // red  
  
$weathers = array('sunny', 'cloudy', 'raining');  
echo $weathers[0]; // sunny
```

# array

```
$fruits = ['apple' => 'red', 'lemon' => 'green', 'pineapple' => 'yellow'];  
echo $fruits['apple']; // red
```

```
$weathers = ['sunny', 'cloudy', 'raining'];  
echo $weathers[0]; // sunny
```

# looping through array

```
$fruits = ['apple' => 'red', 'lemon' => 'green', 'pineapple' => 'yellow'];  
  
echo '我最愛吃的水果是：';  
echo '<ul>';  
foreach ($fruits as $key => $value) {  
    echo "<li>$key</li>";  
}  
echo '</ul>';
```

我最愛吃的水果是：

- apple
- lemon
- pineapple

# looping through array

```
$fruits = ['apple' => 'red', 'lemon' => 'green', 'pineapple' => 'yellow'];  
  
echo '我最愛吃的水果是：';  
echo '<ul>';  
foreach ($fruits as $key => $value) {  
    echo "<li>$key</li>";  
}  
echo '</ul>';
```

# looping through array

```
$fruits = ['apple' => 'red', 'lemon' => 'green', 'pineapple' => 'yellow'];  
  
echo '我最愛吃的水果是：';  
echo '<ul>';  
foreach ($fruits as $fruit => $color) {  
    echo "<li>$fruit</li>";  
}  
echo '</ul>';
```



# modify array

```
$fruits = ['apple' => 'red', 'lemon' => 'green', 'pineapple' => 'yellow'];  
  
$fruits['apple'] = 'green';  
echo $fruits['apple']; // green  
  
unset($fruits['lemon']);  
var_dump($fruits);      // ['apple' => 'red', 'pineapple' => 'yellow']
```

# insert

```
$fruits = [];  
$fruits['apple'] = 'red';  
echo $fruits['apple']; // red
```

# insert

```
$fruits = [];  
$fruits[] = 'apple';  
$fruits[] = 'orange';  
echo $fruits[0]; // apple
```

# modify array

```
$fruits = ['apple', 'lemon', 'pineapple'];
```

```
$menu = implode(',', $fruits);
```

```
echo 'We have '.$menu;
```

```
// We have apple, lemon, pineapple
```

```
$transport = 'car, bus, MRT, airplane';
```

```
$transport = explode(',', $transport);
```

```
var_dump($transport);
```

```
// ['car', 'bus', 'MRT', 'airplane']
```

# sort

```
$fruits = ['pineapple', 'lemon', 'apple'];  
$fruitsColor = [  
    'apple' => 'red',  
    'lemon' => 'green',  
    'pineapple' => 'yellow'  
];  
  
sort($fruits);  
sort($fruitsColor);  
  
var_dump($fruits);  
// ['apple', 'lemon', 'pineapple']  
var_dump($fruitsColor);  
// ['green', 'red', 'yellow']
```

# asort

```
$fruitsColor = [  
    'apple' => 'red',  
    'lemon' => 'green',  
    'pineapple' => 'yellow'  
];  
  
asort($fruitsColor);  
  
var_dump($fruitsColor);  
// ['lemon'=>'green', 'apple'=>'red', 'pineapple'=>'yellow']
```

# ksort

```
$fruitsColor = [  
    'pineapple' => 'yellow',  
    'apple' => 'red',  
    'lemon' => 'green',  
];  
  
ksort($fruitsColor);  
  
var_dump($fruitsColor);  
// ['apple'=>'red', 'lemon'=>'green', 'pineapple'=>'yellow']
```

# rsort, arsort, krsort

reverse the order



# function

groups of logic

# function

```
function isNight($time) {  
    if ($time > 18)  
        return true;  
    else  
        return false;  
}
```

```
$isNight = isNight(20);  
if ($isNight)  
    echo 'turn on the light';  
else  
    echo 'turn off the light';
```

# function

```
function isNight($time) {  
    if ($time > 18)  
        return true;  
    else  
        return false;  
}  
  
$isNight = isNight(20);  
if ($isNight)  
    echo 'turn on the light';  
else  
    echo 'turn off the light';
```

# scope

```
$transport = '機車';
```

```
function rainingDayTransport() {  
    echo '雨天就是要搭';  
    $transport = 'MRT';  
    echo $transport;  
    echo "\n";  
}
```

```
function tooFatTransport() {  
    echo '吃太好也許可以';  
    $transport = '走路';  
    echo $transport;  
    echo "\n";  
}
```

```
rainingDayTransport();  
tooFatTransport();  
echo '平常出門騎';  
echo $transport;  
echo "\n";
```

# object

data & logic together

# object

```
class Computer {  
    public $cpu;  
    public $ram;  
    public $hd;  
    public $power;  
  
    public function turnOn() {  
        $this->power = true;  
    }  
  
    public function turnOff() {  
        $this->power = false;  
    }  
}
```

```
$mac = new Computer;
```

# object

```
class Computer {  
    public $cpu;  
    public $ram;  
    public $hd;  
    public $power;  
  
    public function turnOn() {  
        $this->power = true;  
    }  
  
    public function turnOff() {  
        $this->power = false;  
    }  
}
```

```
$mac = new Computer;
```

# object

```
class Doggy {  
    public $color;  
  
    public function bark() {  
        echo '汪汪!';  
    }  
}
```

```
$charlie = new Doggy;  
$charlie->color = 'yellow';  
$charlie->bark();
```



# constructor

```
class Computer {  
    public $cpu;  
    public $ram;  
    public $hd;  
    public $power;
```

```
    public function __construct($cpu, $ram, $hd) {  
        $this->cpu = $cpu;  
        $this->ram = $ram;  
        $this->hd = $hd;  
  
        $this->power = false;  
    }
```

```
    public function turnOn() {  
        $this->power = true;  
    }
```

```
    public function turnOff() {  
        $this->power = false;  
    }
```

```
}
```

```
$mac = new Computer('1.2GHz', '2GB', '2TB');  
echo $mac->cpu;  
echo $mac->ram;  
echo $mac->hd;
```

# extending an object

```
class ATM extends Computer {  
    public $monitor;  
    public $keyboard;  
}  
  
$bankATM = new ATM('1.2GHz', '2GB', '2TB');
```

# property & method visibility

```
class Computer {  
    private $branding;  
    protected $power;  
    public $cpu;  
    public $ram;  
    public $hd;  
  
    public function __construct($cpu, $ram, $hd) {  
        $this->cpu = $cpu;  
        $this->ram = $ram;  
        $this->hd = $hd;  
  
        $this->power = false;  
    }  
  
    public function turnOn() {  
        $this->power = true;  
    }  
  
    public function turnOff() {  
        $this->power = false;  
    }  
}
```

```
$mac = new Computer('1.2GHz', '2GB', '2TB');  
echo $mac->cpu;  
echo $mac->ram;  
echo $mac->hd;  
$mac->power = 123; // error  
$mac->turnOn();
```

# property & method visibility

```
class Computer {  
    private $branding;  
  
    public function setBranding($branding) {  
        $this->branding = $branding.' ©';  
    }  
  
    public function getBranding() {  
        return $this->branding;  
    }  
}
```

```
$mac = new Computer;  
$mac->setBranding('Apple');  
echo $mac->getBranding();    // 'Apple ©'
```

# private & protected

```
class Computer {  
    private $branding;  
    protected $power;  
  
    public function __construct() {  
        $this->turnOff();  
    }  
  
    public function turnOn() {  
        $this->power = true;  
    }  
  
    public function turnOff() {  
        $this->power = false;  
    }  
}
```

```
class ATM extends Computer {  
    public $monitor;  
    public $keyboard;  
  
    public function isPowerOn() {  
        return $this->power;  
    }  
}
```

```
$bankATM = new ATM;  
$bankATM->turnOn();  
echo $bankATM->isPowerOn();
```



# private & protected

```
class Computer {  
    private $branding;  
  
    public function setBranding($branding) {  
        $this->branding = $branding.' ©';  
    }  
  
    public function getBranding() {  
        return $this->branding;  
    }  
}
```

```
class ATM extends Computer {  
    public $monitor;  
    public $keyboard;  
  
    public function getBrandingName() {  
        return $this->branding;  
    }  
}
```

```
$bankATM = new ATM;  
$bankATM->setBranding('IBM');  
echo $bankATM->getBranding();  
echo $bankATM->getBrandingName();
```

# namespace

**class ~~Vehicle~~**

**class Vehicle**

# namespace

\你家的\小黑

\我家的\小黑



# namespace

```
namespace Taker;

class Say {
    public function hi() {
        echo 'Hello, my friend!';
    }
}
```

```
include 'Say.php';

\Taker\Say::hi();
```

# namespace

```
namespace Taker;
```

```
class Say {  
    public function hi() {  
        echo 'Hello, my friend!';  
    }  
}
```

```
namespace Taker\Classroom;
```

```
class Say {  
    public function welcome() {  
        echo 'Welcome to my classroom!';  
    }  
}
```

```
include 'Say.php';  
include 'Classroom.php';
```

```
\Taker\Say::hi();  
\Taker\Classroom\Say::welcome();
```

# namespace + use

```
namespace Taker;
```

```
class Say {  
    public function hi() {  
        echo 'Hello, my friend!';  
    }  
}
```

```
namespace Taker\Classroom;
```

```
class Say {  
    public function welcome() {  
        echo 'Welcome to my classroom!';  
    }  
}
```

```
include 'Say.php';  
include 'Classroom.php';
```

```
use \Taker\Say;  
use \Taker\Classroom\Say as Classroom;
```

```
Say::hi();  
Classroom::welcome();
```

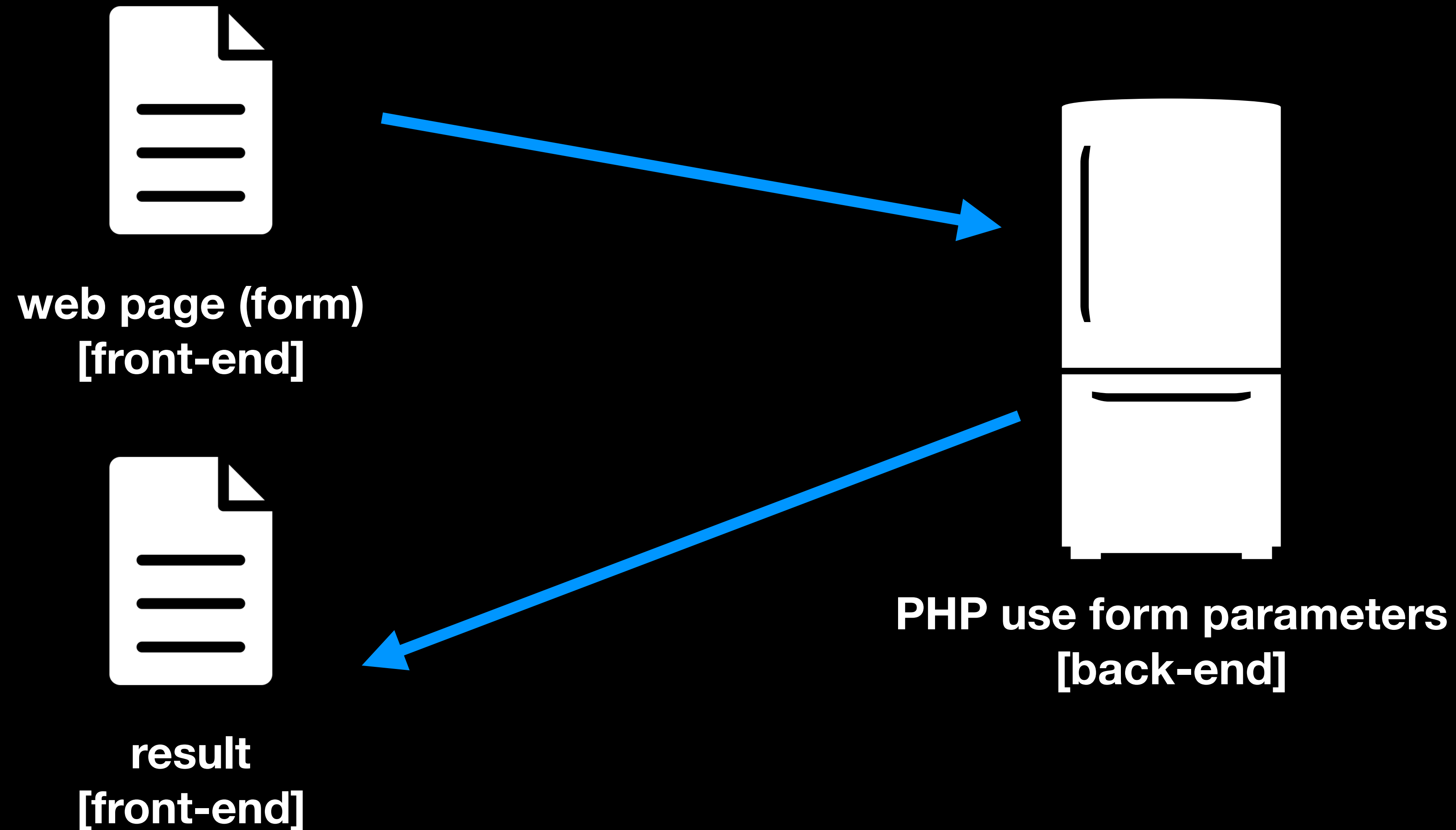
# object

data & logic together

# web form & server variables

exchanging info with users

# web form & server variables



Edit your profile



Full name

Taker Wu

Display name (optional)

taker

This could be your first name, or a nickname — however you'd like people to refer to you in Slack.

What I do

What I do

Let people know what you do at 網頁基礎15天.

Status

☺ What's your status?

Phone number

(123) 555-5555

Enter a phone number.

Time Zone

(UTC+08:00) Taipei ▼

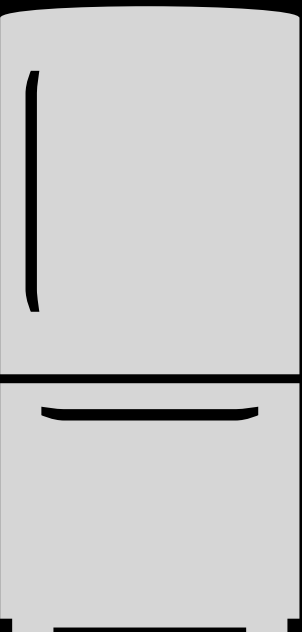
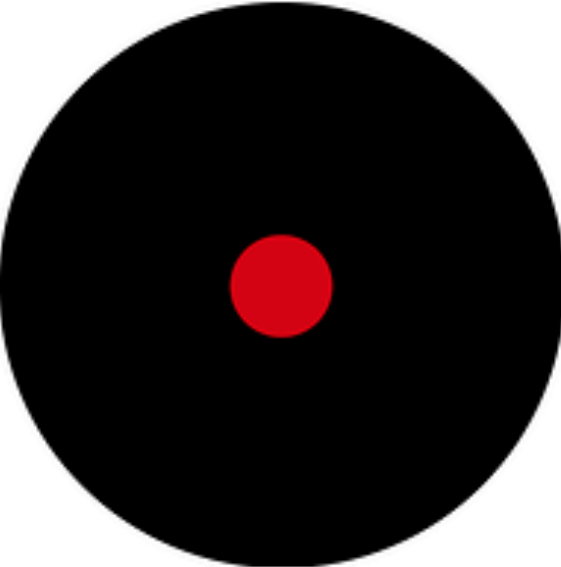
Your current time zone. Used to send summary and notification emails, for times in your activity feeds, and for reminders.

Skype

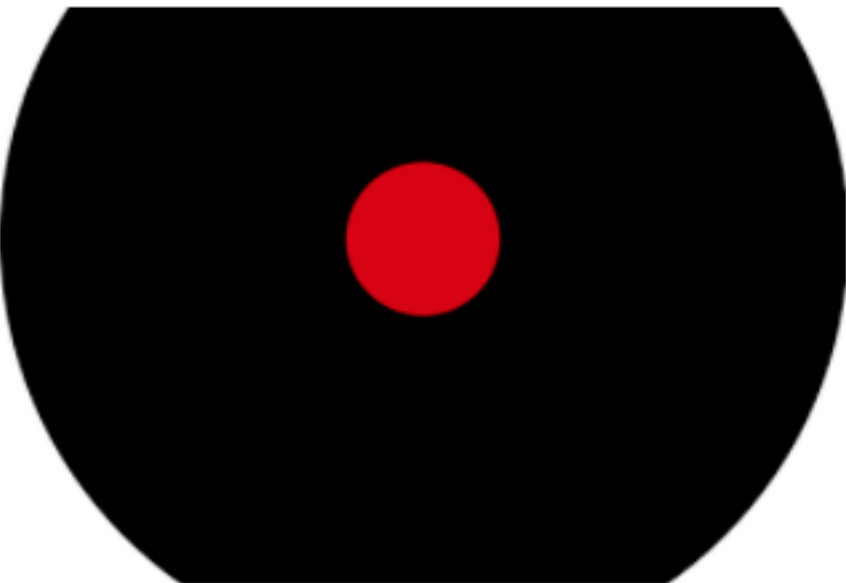
Skype

This will be displayed on your profile.

Profile photo



< Workspace Directory



Taker Wu ●

Edit Profile



Status ☺ What's your status?

Display name	taker
Timezone	11:06 AM local time (change)
Email	taker.wu@gmail.com
Role	Owner of 網頁基礎15天

Cancel

Save Changes

# form (front-end)

```
<form action="update-account-info.php" method="post">  
  Name: <input type="text" name="name"><br>  
  Email: <input type="email" name="email"><br>  
  
  Gender: <br>  
  <input type="radio" name="gender" value="male" checked> Male<br>  
  <input type="radio" name="gender" value="female"> Female<br>  
  
  <button>submit</button>  
</form>
```

form elements  
[https://www.w3schools.com/html/html\\_form\\_elements.asp](https://www.w3schools.com/html/html_form_elements.asp)

Name:

Email:

Gender:

☒ Male

☐ Female



# accessing form parameters (back-end)

```
echo $_POST['name'];  
echo $_POST['email'];  
echo $_POST['gender'];
```

# form (front-end)

```
<form action="update-account-info.php" method="post">  
  Name: <input type="text" name="name"><br>  
  Email: <input type="email" name="email"><br>  
  
  Gender: <br>  
  <input type="radio" name="gender" value="male" checked> Male<br>  
  <input type="radio" name="gender" value="female"> Female<br>  
  
  <button>submit</button>  
</form>
```

update-account-info.php?name=John&email=john@gmail.com&gender=male

# form (front-end)

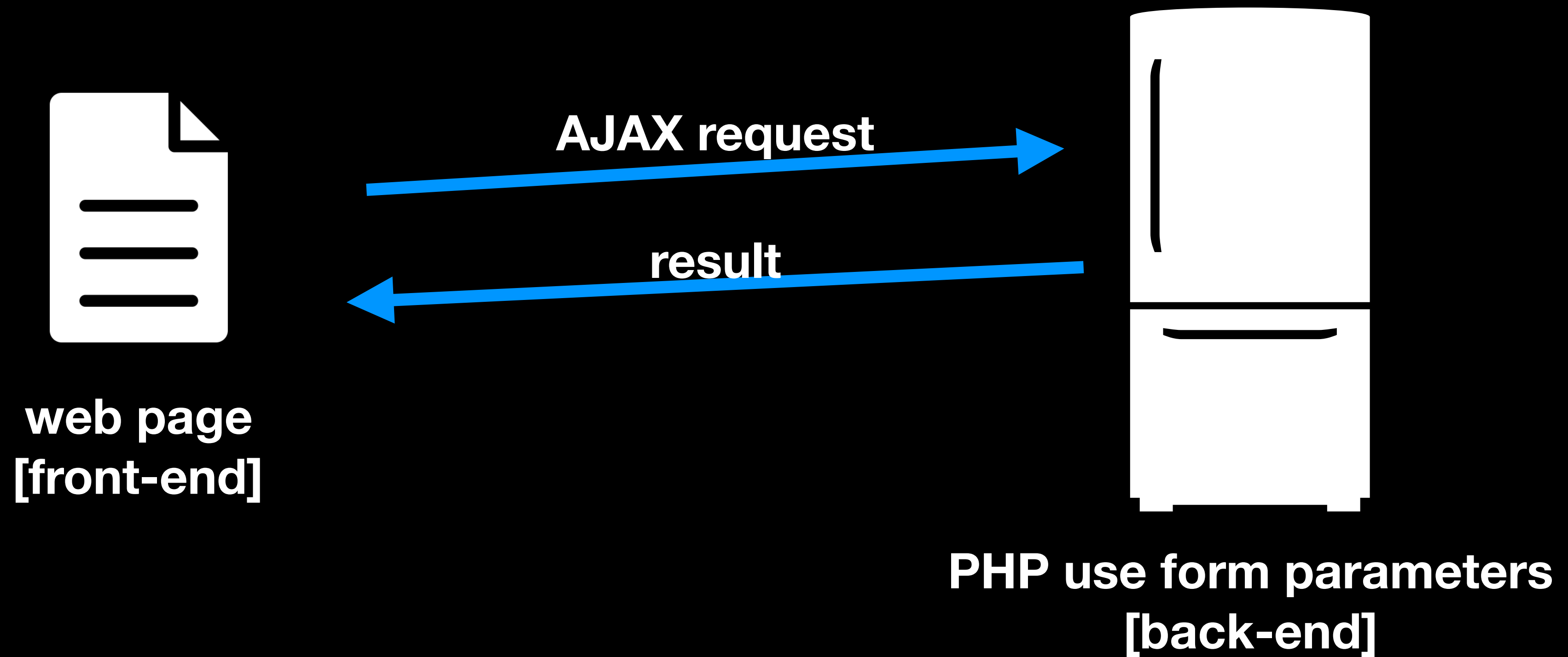
```
<form action="update-account-info.php" method="post">  
  Name: <input type="text" name="name"><br>  
  Email: <input type="email" name="email"><br>  
  
  Gender: <br>  
  <input type="radio" name="gender" value="male" checked> Male<br>  
  <input type="radio" name="gender" value="female"> Female<br>  
  
  <button>submit</button>  
</form>
```

update-account-info.php

# accessing form parameters (back-end)

```
echo $_POST['name'];  
echo $_POST['email'];  
echo $_POST['gender'];  
  
echo $_GET['name'];  
echo $_GET['gender'];  
  
echo $_REQUEST['name'];  
echo $_REQUEST['gender'];
```

# AJAX





# AJAX

## 購買清單

0 Completed

☐ 牛奶

☐ 水梨

☐ 香蕉

☐ 欲飯糰

☐



New Item...

# validating

```
if (filter_var($_POST['email'], FILTER_VALIDATE_EMAIL)) {  
    echo "Email OK!";  
}  
else {  
    echo "Email 有錯!";  
}
```

filter\_var

<http://php.net/manual/en/function.filter-var.php>

filters

<http://php.net/manual/en/filter.filters.validate.php>

# server variables (back-end)

```
echo $_SERVER['QUERY_STRING']; // section=1&course=20
echo $_SERVER['PATH_INFO']; // /classroom
echo $_SERVER['SERVER_NAME']; // www.everyonecanwebsite.com
echo $_SERVER['DOCUMENT_ROOT']; // /usr/local/htdocs
echo $_SERVER['REMOTE_ADDR']; // 175.38.221.3
echo $_SERVER['HTTP_REFERER']; // https://www.facebook.com
echo $_SERVER['HTTP_USER_AGENT'];
// Mozilla/5.0 (Windows NT 5.1; rv:12.0) Gecko/20100101 Firefox/12.0
```



# database

remembering info

# PDO

PHP Data Object

# connect to database program

```
$user = 'taker';  
$pass = 'passwordjustfortest';  
  
$db = new PDO('mysql:host=localhost;dbname=myblog', $user, $pass);
```

# insert data

```
$user = 'taker';  
$password = 'testingpassword';  
$pdo = new PDO('mysql:host=localhost;dbname=my_blog', $user, $password);  
  
$affectedRows = $pdo->exec('  
    INSERT INTO posts (author_id, category_id, title, content, post_time)  
    VALUES (1, 10, "10 things you should know about MySQL", "...", "2017-10-20 17:00:00")  
');
```

# retrieve data

```
$user = 'taker';  
$password = 'testingpassword';  
$pdo = new PDO('mysql:host=localhost;dbname=my_blog', $user, $password);  
  
$query = $pdo->query('SELECT author_id, title FROM posts');  
while ($row = $query->fetch()) {  
    echo $row['title'].'<br>';  
}
```

# security issue

```
$pdo = new PDO('mysql:host=localhost;dbname=my_blog', $user, $pass);

$sql = 'SELECT * FROM posts WHERE author_id=:author_id';
$stmt = $pdo->prepare($sql);
$stmt->bindValue(':author_id', $_POST['author_id']);
$stmt->execute();

while (($result = $stmt->fetch(PDO::FETCH_ASSOC)) !== false) {
    echo $result['title'];
}
```

# security issue



# security issue

```
$user = 'taker';  
$password = 'testingpassword';  
$pdo = new PDO('mysql:host=localhost;dbname=my_blog', $user, $password);  
  
$query = $pdo->query('SELECT author_id, title FROM posts WHERE id='.$_POST['id']);  
while ($row = $query->fetch()) {  
    echo $row['title'].'<br>';  
}
```



# security issue

```
$pdo = new PDO('mysql:host=localhost;dbname=my_blog', $user, $password);

$sql = 'SELECT * FROM posts WHERE author_id=:author_id';
$stmt = $pdo->prepare($sql);
$stmt->bindValue(':author_id', $author_id);
$stmt->execute();

while ($row = $stmt->fetch(PDO::FETCH_ASSOC)) {
    echo htmlentities($row['title']). '<br />';
}
```

htmlentities

<http://php.net/manual/en/function.htmlentities.php>

htmlspecialchars

<http://php.net/manual/en/function htmlspecialchars.php>

比較

<http://lifeiskuso.blogspot.tw/2009/08/php-htmlentities-htmlspecialchars.html>

# security issue

slash	&#47;	&frasl;	/
digits 0-9	&#48;- &#57;		
colon	&#58;		:
semicolon	&#59;		;
less-than sign	&#60;	&lt;	<
equals sign	&#61;		=
greater-than sign	&#62;	&gt;	>
question mark	&#63;		?
at sign	&#64;		@
uppercase letters A-Z	&#65;- &#90;		

html special character  
<http://www.degraeve.com/reference/specialcharacters.php>

# fetch

- fetch
- fetchAll
- PDO::FETCH\_ASSOC

<http://php.net/manual/en/pdostatement.fetch.php>

# fetch

```
$pdo = new PDO('mysql:host=localhost;dbname=my_blog', $user, $pass);

$sql = 'SELECT * FROM posts WHERE author_id=:author_id';
$stmt = $pdo->prepare($sql);
$stmt->bindValue(':author_id', $_POST['author_id']);
$stmt->execute();

while (($result = $stmt->fetch(PDO::FETCH_ASSOC)) !== false) {
    echo $result['title'];
}
```

# fetch

```
$pdo = new PDO('mysql:host=localhost;dbname=my_blog', $user, $pass);

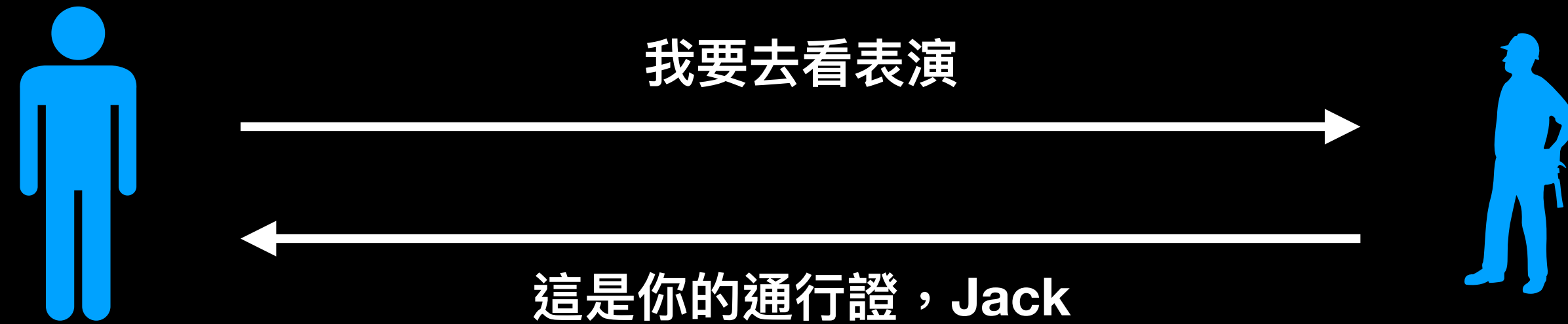
$sql = 'SELECT * FROM posts WHERE author_id=:author_id';
$stmt = $pdo->prepare($sql);
$stmt->bindValue(':author_id', $_POST['author_id']);
$stmt->execute();

$results = $stmt->fetchAll(PDO::FETCH_ASSOC);
foreach ($results as $key => $result) {
    echo $result['title'];
}
```

# cookies & sessions

remembering users

# cookies & sessions

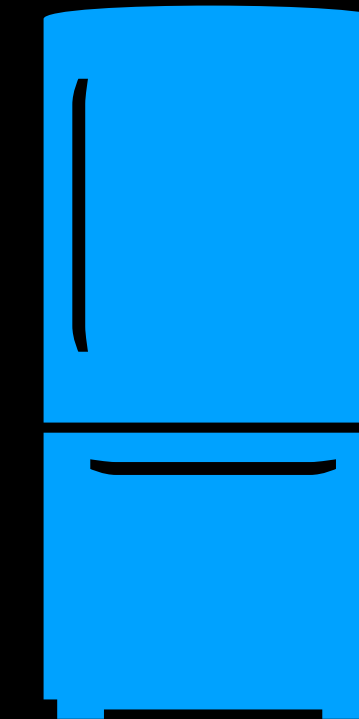




# cookie



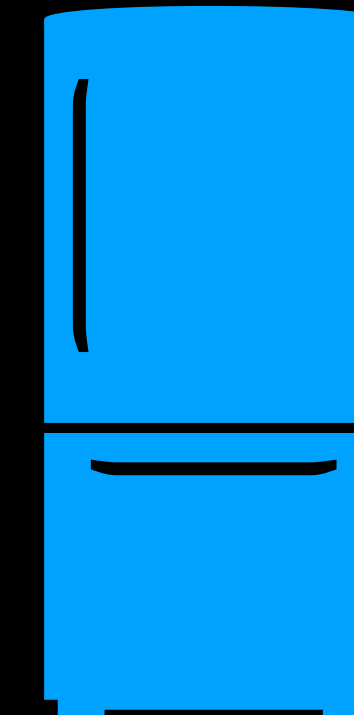
I wanna shopping



OK, and let me call you "Jack"



Hi, I'm Jack, show me more products



OK Jack, these products suit for you



# cookie

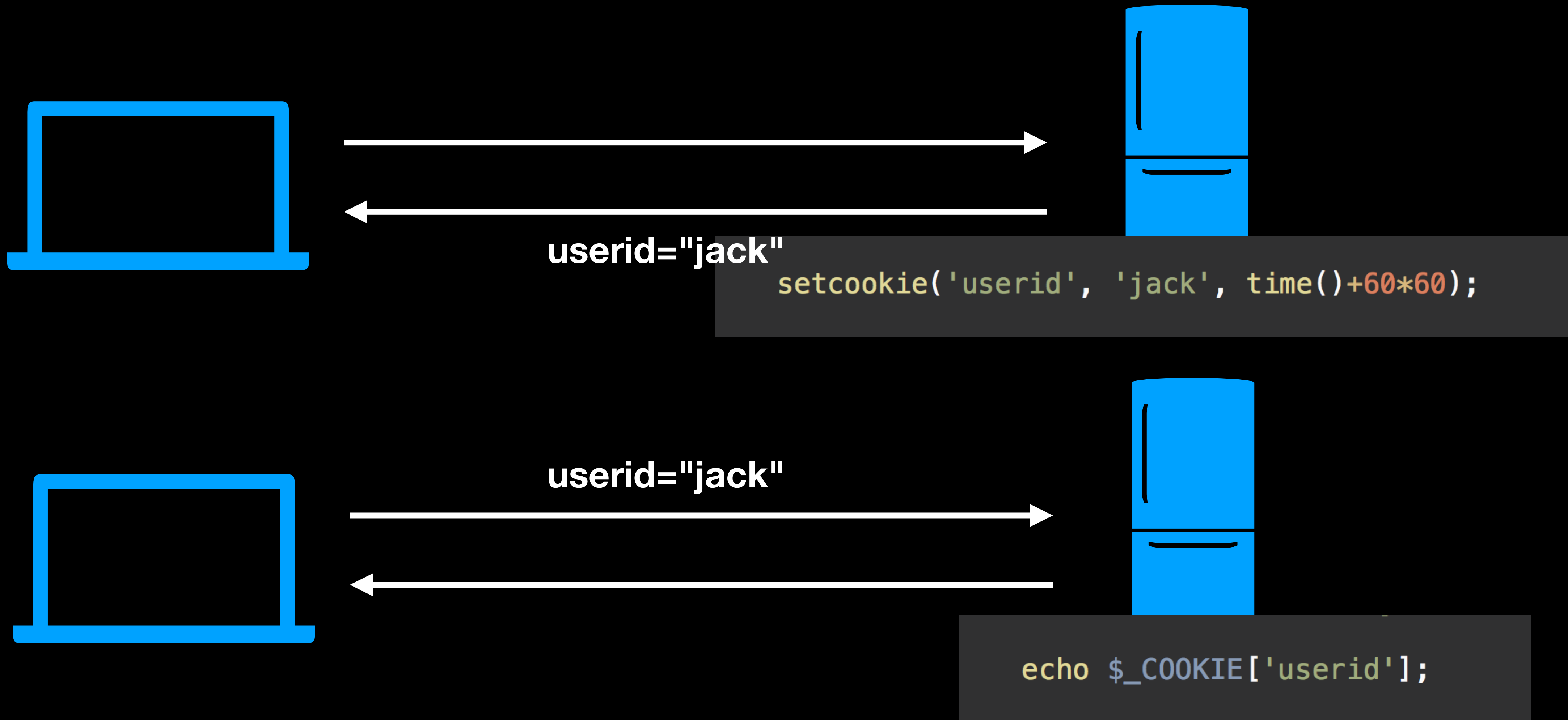
request 1

```
setcookie('userid', 'jack', time()+60*60);
```

request 2

```
echo $_COOKIE['userid'];
```

# cookie



# session



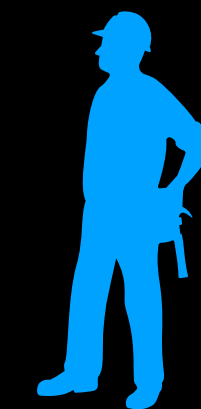
我想申辦 VIP，這是我的資料



你的 VIP 編號是 9487



你好，我是 9487，我要看一下我的資料



9487，你的名字是 Jack、電話是。。。

# session

```
session_start();
```

```
if (!isset($_SESSION['user_id']))  
    $_SESSION['user_id'] = 'Jack';
```

```
echo 'hello, ' . $_SESSION['user_id'];
```

# session

```
ini_set('session.gc_maxlifetime', 3600);  
session_start();  
  
if (!isset($_SESSION['user_id']))  
    echo 'you are not logged in.';  
else  
    echo 'hello, ' . $_SESSION['user_id'];
```

# login

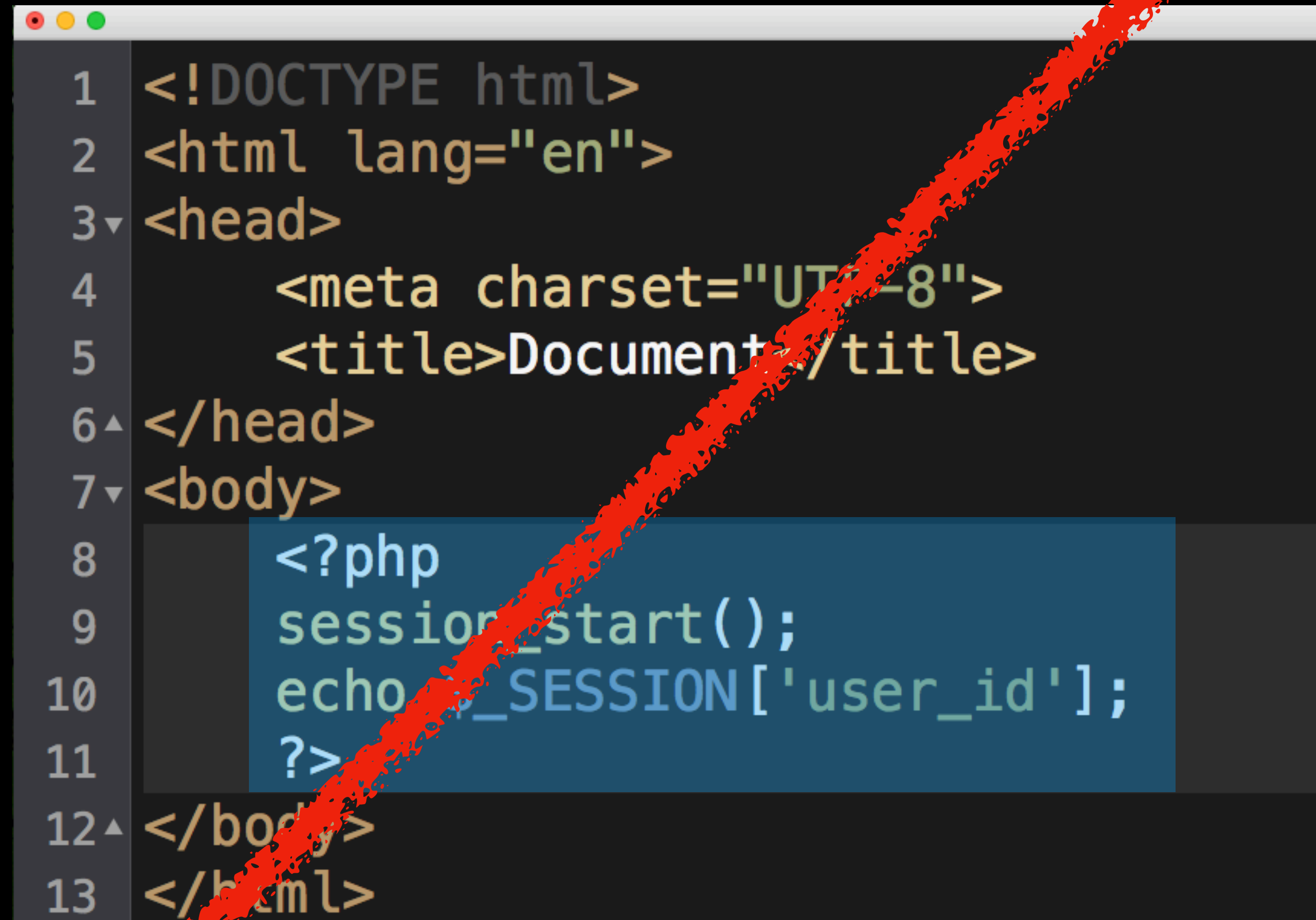
- user 透過登入頁面填寫 username/password
  - server 收到 username/password 之後做驗證
  - 驗證成功，把 username 存到 session 裡面（登入成功）
- user 要求看 VIP 的頁面
  - server 可以透過 session，取得相對應的資料給 user 看
- user 要把電腦還朋友了，按下登出
  - 從 session 清除 username 資訊（登出）

# header already sent

```
1 <?php
2
3 setcookie('userid', 'jack');
4 echo 'hello, world';
5
6
```

```
1 <?php
2
3 session_start();
4 echo $_SESSION['user_id'];
5
6
```

# header already sent



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>Document</title>
6 </head>
7 <body>
8     <?php
9     session_start();
10    echo $_SESSION['user_id'];
11    ?>
12 </body>
13 </html>
```



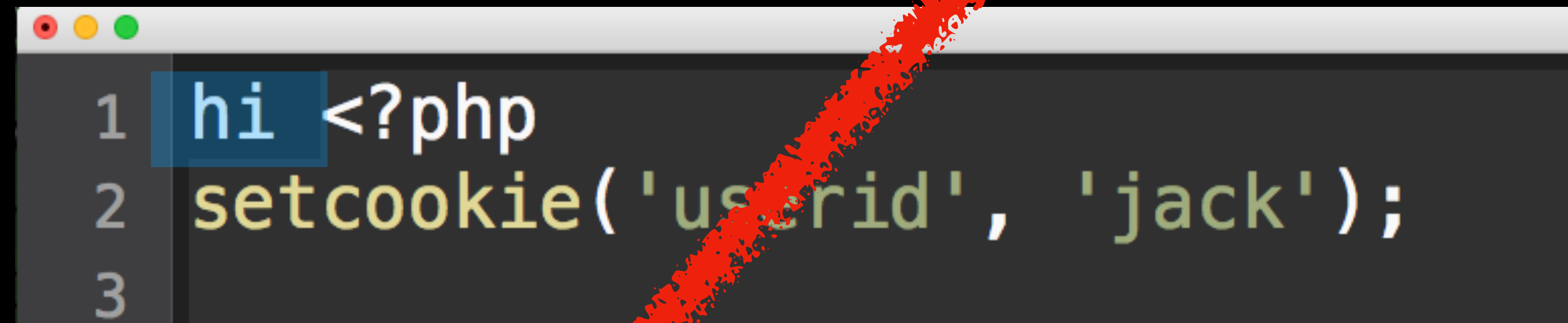
# header already sent



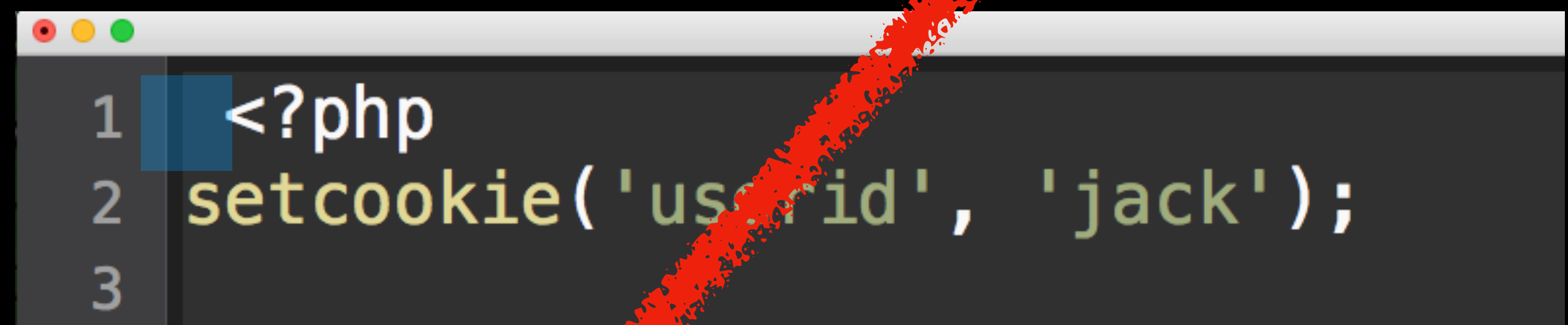
```
1 <?php
2 echo 'hi';
3 setcookie('userid', 'jack');
4
```

The image shows a code editor window with a dark background. The code is written in PHP and consists of four lines. The first line is a PHP opening tag. The second line is an echo statement. The third line is a setcookie function call. The fourth line is empty. A red diagonal line is drawn across the code, indicating an error. The error message 'header already sent' is visible at the top of the image.

# header already sent



```
1 hi <?php
2 setcookie('userid', 'jack');
3
```



```
1 <?php
2 setcookie('userid', 'jack');
3
```

# header



header

output

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  ...
</body>
</html>
```

# output

```
1 <?php
2 echo 'hi';
3 setcookie('userid', 'jack');
4
```

```
1 hi <?php
2 setcookie('userid', 'jack');
3
```

```
1 <?php
2 setcookie('userid', 'jack');
3
```

# output

header

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>Document</title>
6 </head>
7 <body>
8     <?php
9     session_start();
10    echo $_SESSION['user_id'];
11    ?>
12 </body>
13 </html>
```

# header

## ▼ General

**Request URL:** https://www.everyonecanwebsite.com/

**Request Method:** GET

**Status Code:** 🟢 200

**Remote Address:** 104.27.159.166:443

**Referrer Policy:** no-referrer-when-downgrade

## ▼ Response Headers

**cache-control:** no-store, no-cache, must-revalidate, post-check=0, pre-check=0

**cf-ray:** 3ae9ebac3de032b9-HKG

**content-encoding:** gzip

**content-type:** text/html; charset=UTF-8

**date:** Mon, 16 Oct 2017 09:16:40 GMT

**expires:** Thu, 19 Nov 1981 08:52:00 GMT

**pragma:** no-cache

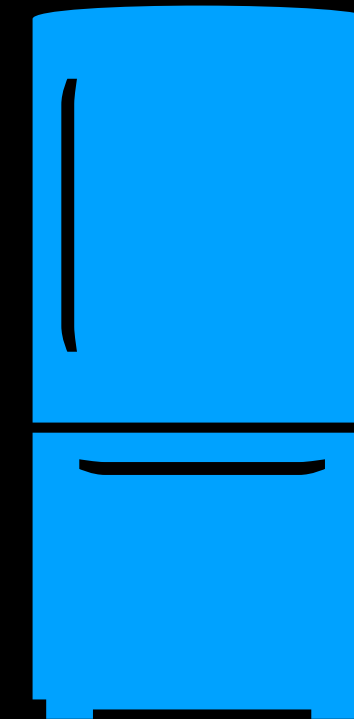
**server:** cloudflare-nginx

**set-cookie:** \_identity=5e5b18006a02d969bc1b26c43b20a38b99a71b279bf061fa4213d3656aceb072a%3A%22\_identity%22%3Bi%3A1%3Bs%3A17%3A%22%5B1%2Cnull%2C31104000%5D%22%3B%7D; expires=Thu, MT; Max-Age=31104000; path=/; httponly

**status:** 200

**vary:** Accept-Encoding

# header



header

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  ...
</body>
</html>
```

# header already sent

header

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <?php setcookie('userid', 'jack'); ?>
</body>
</html>
```

A white arrow points upwards from the PHP code line in the body section to the blue header bar.



# header already sent

```
1 <?php
2
3 setcookie('userid', 'jack');
4 echo 'hello, world';
5
6
```

```
1 <?php
2
3 session_start();
4 echo $_SESSION['user_id'];
5
6
```

# cURL

talking to other website / API

# cURL

```
$formData = [  
    'name' => 'Taker',  
    'email' => 'taker.wu@everyonecanwebsite.com',  
];  
  
$c = curl_init('http://www.example.com/test.php');  
curl_setopt($c, CURLOPT_RETURNTRANSFER, true);  
curl_setopt($c, CURLOPT_POST, true);  
curl_setopt($c, CURLOPT_POSTFIELDS, $formData);  
  
$serverOutput = curl_exec ($c);  
echo $serverOutput;  
  
curl_close ($c);
```

# composer

package management

# install composer

- Mac
- Windows

# Mac

## Download Composer

Run this in your terminal to get the latest Composer version:

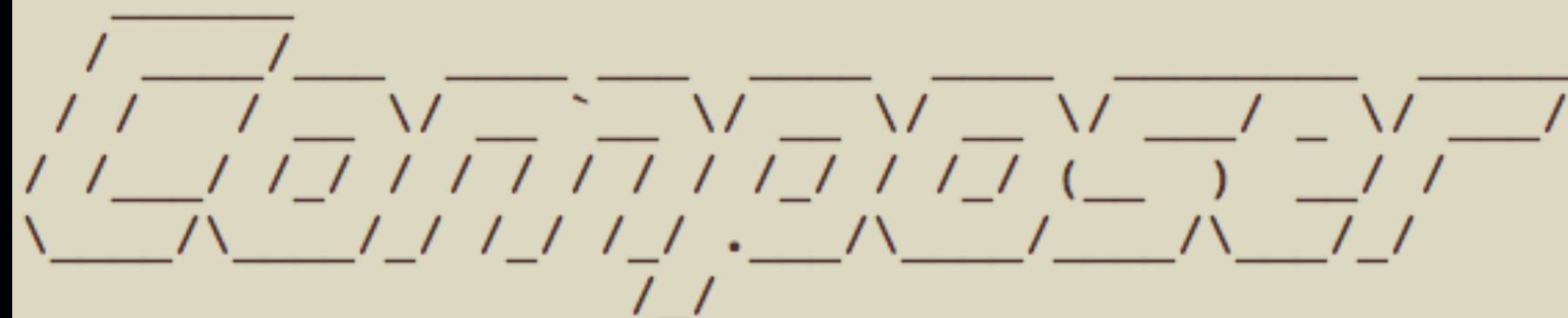
```
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
php -r "if (hash_file('SHA384', 'composer-setup.php') === '544e09ee996cdf60ece3804a
php composer-setup.php
php -r "unlink('composer-setup.php');"
```

This installer script will simply check some php.ini settings, warn you if they are set incorrectly, and then download the latest composer.phar in the current directory. The 4 lines above will, in order:

- Download the installer to the current directory
- Verify the installer SHA-384 which you can also [cross-check here](#)
- Run the installer
- Remove the installer

# Mac

php bin/composer



Composer version 1.4.1 2017-03-10 09:29:45

## Usage:

command [options] [arguments]

## Options:

-h, --help	Display this help message
-q, --quiet	Do not output any message
-V, --version	Display this application version
--ansi	Force ANSI output
--no-ansi	Disable ANSI output
-n, --no-interaction	Do not ask any interactive question
--profile	Display timing and memory usage information
--no-plugins	Whether to disable plugins.
-d, --working-dir=WORKING-DIR	If specified, use the given directory as working directory.
-v vv vvv, --verbose	Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug

# Globally

~~composer~~

```
mv composer.phar /usr/local/bin/composer
```

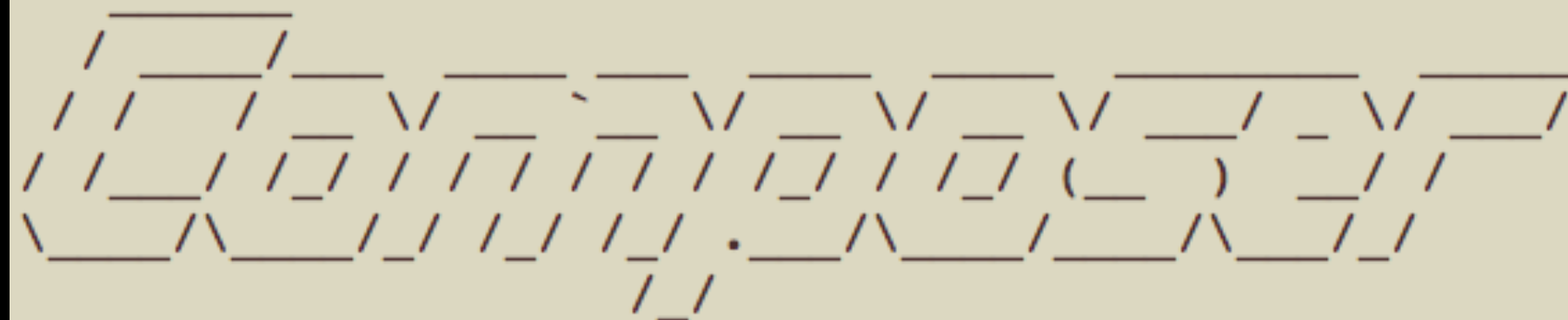
composer



# Windows

download Setup.exe

composer



Composer version 1.4.1 2017-03-10 09:29:45

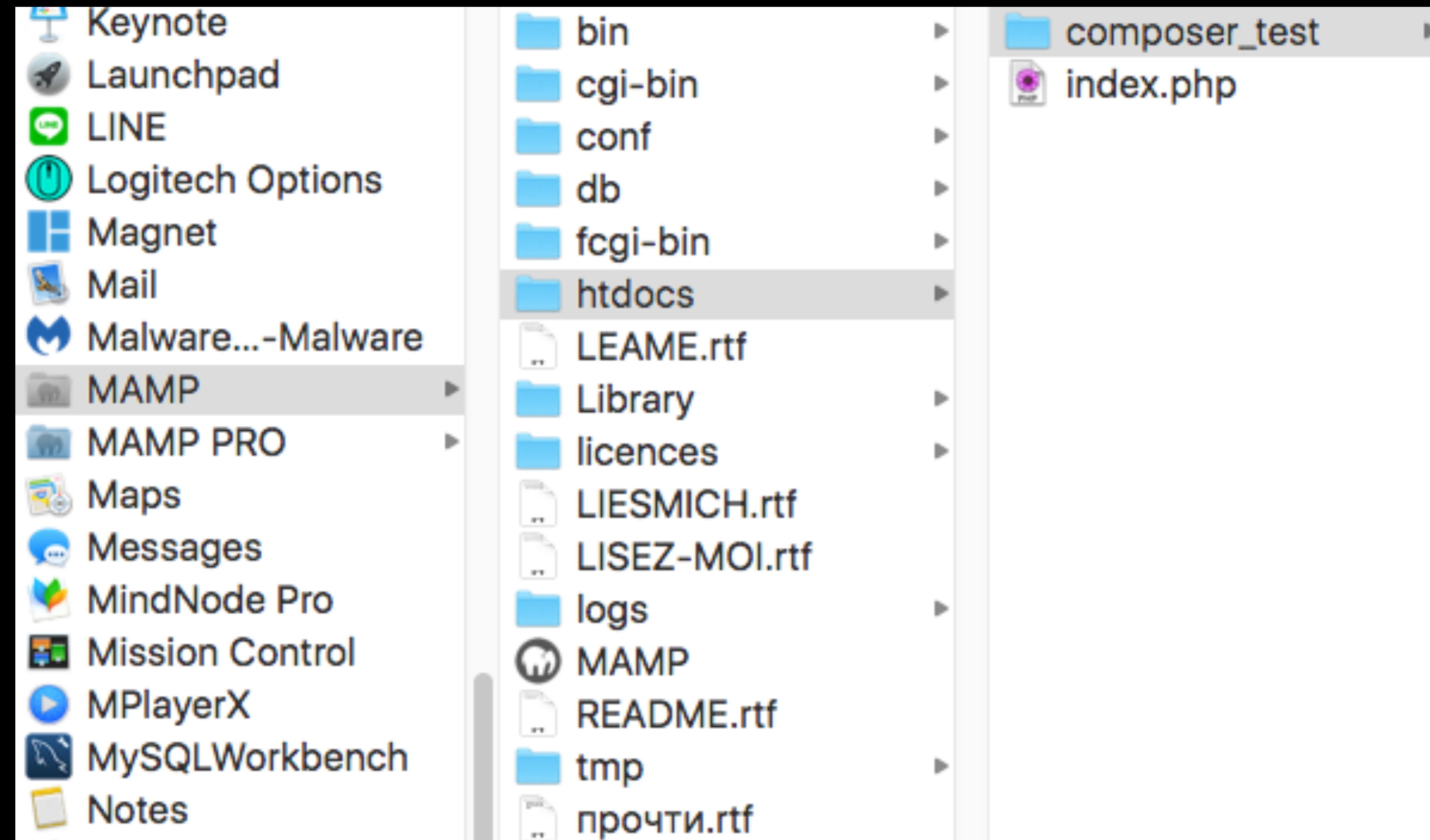
## Usage:

command [options] [arguments]

## Options:

-h, --help	Display this help message
-q, --quiet	Do not output any message
-V, --version	Display this application version
--ansi	Force ANSI output
--no-ansi	Disable ANSI output
-n, --no-interaction	Do not ask any interactive question
--profile	Display timing and memory usage information
--no-plugins	Whether to disable plugins.
-d, --working-dir=WORKING-DIR	If specified, use the given directory as working directory.
-v vv vvv, --verbose	Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug

# adding package to your program



# adding package to your program

```
cd /Applications/MAMP/htdocs/composer_test
```

```
composer require kint-php/kint
```

```
Takerde-MacBook-Pro:composer_test taker$ composer require kint-php/kint
Using version ^2.2 for kint-php/kint
./composer.json has been created
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
  - Installing kint-php/kint (2.2): Downloading (100%)
Writing lock file
Generating autoload files
Takerde-MacBook-Pro:composer_test taker$
```



composer.json



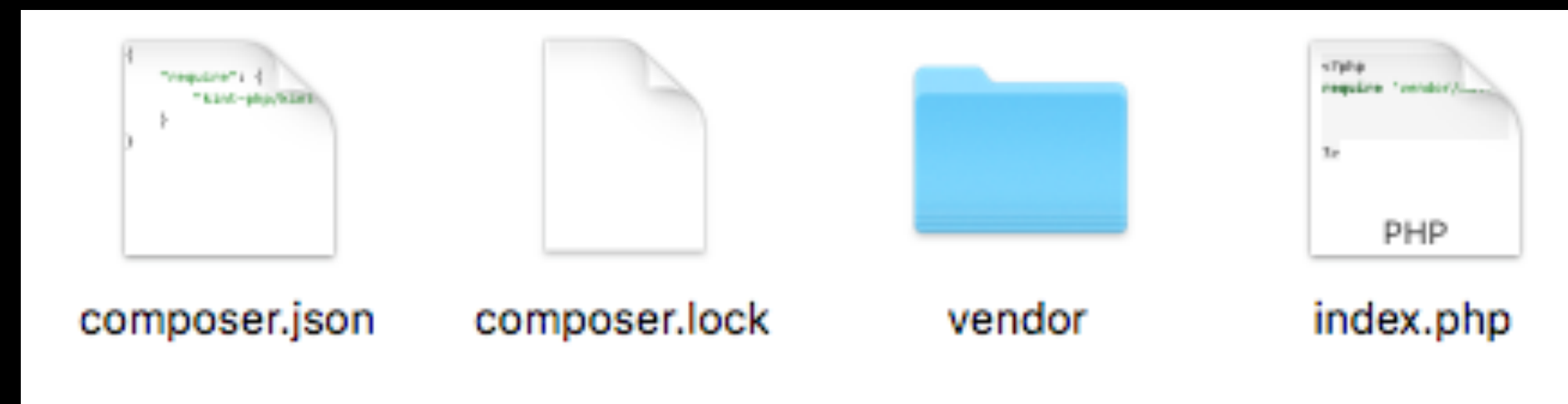
composer.lock



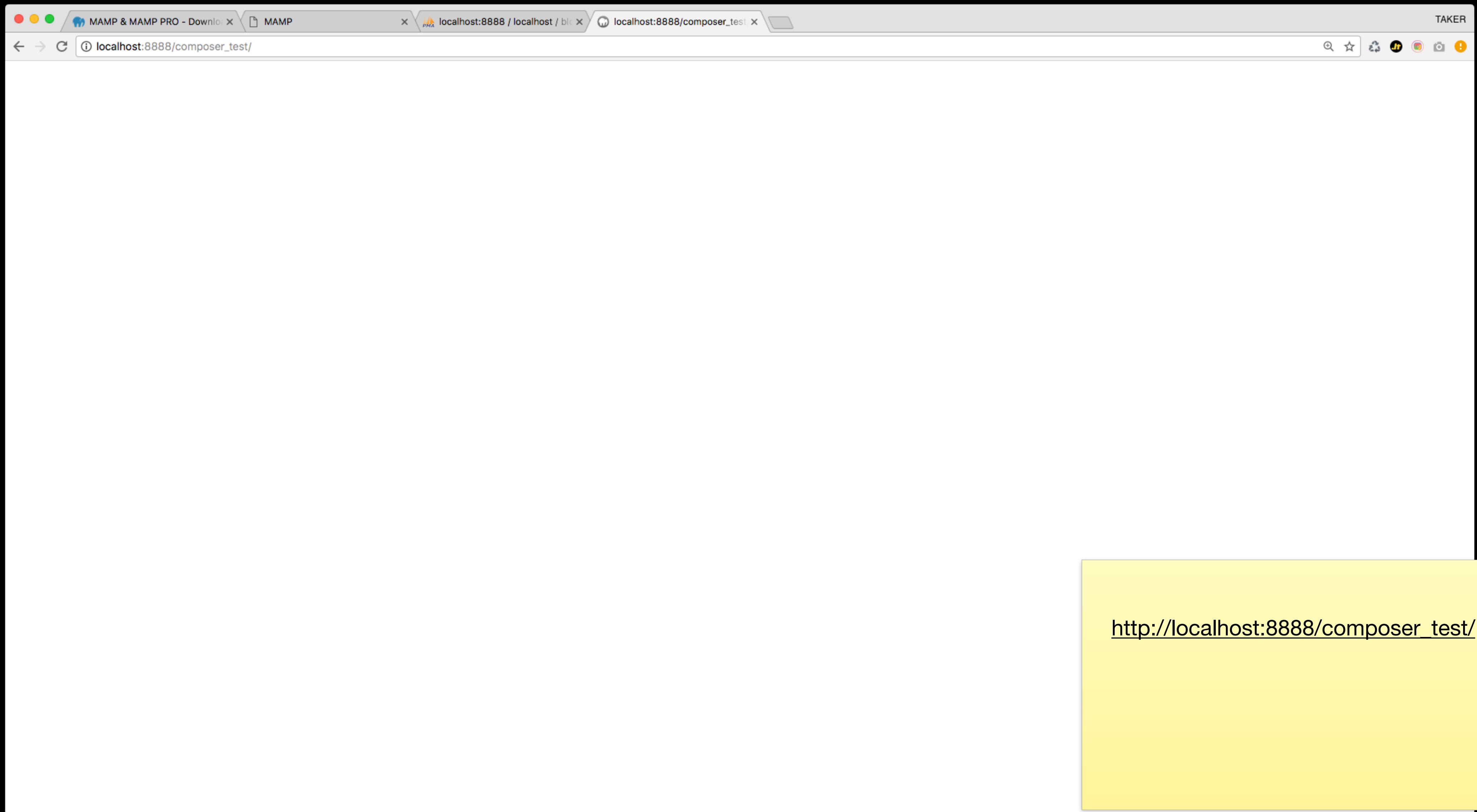
vendor

# adding package to your program

```
<?php  
require 'vendor/autoload.php';  
  
?>
```







# adding package to your program


```
<?php  
require 'vendor/autoload.php';  
  
$foo = [1, 17, "hello", null, [1, 2, 3]];  
d($foo);  
  
?>
```

# adding package to your program

```
[-] $foo array (5)  
  0 => integer 1  
  1 => integer 17  
  2 => string (5) "hello"  
  3 => null  
  [-] 4 => array (3)  
    0 => integer 1  
    1 => integer 2  
    2 => integer 3  
Called from .../index.php:5
```

# finding packages

**Packagist** The PHP Package Repository[Browse](#)[Submit](#)[Create account](#)[Sign in](#)



Packagist is the main [Composer](#) repository. It aggregates public PHP packages installable with Composer.

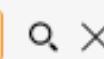
<b>maximebf/debugbar</b> Debug bar in the browser for php application	PHP	↓ 6 410 196 ★ 1 778
<b>barryvdh/laravel-debugbar</b> PHP Debugbar integration for Laravel	PHP	↓ 5 971 716 ★ 5 892
<b>twig/extensions</b> Common additional features for Twig that do not directly belong in core	PHP	↓ 18 831 935 ★ 576
<b>symfony/var-dumper</b> Symfony mechanism for exploring and dumping PHP variables	PHP	↓ 30 594 723 ★ 301
<b>symfony/debug</b> Symfony Debug Component	PHP	↓ 50 211 576 ★ 68
<b>cakephp/debug_kit</b> CakePHP Debug Kit	PHP	↓ 1 797 293 ★ 875
<b>tracy/tracy</b> Tracy: the addictive tool to ease debugging PHP code for cool developers. Friendly design, logging, profiler, advanced features like debugging AJAX calls or CLI support. You will love it.	PHP	↓ 1 447 314 ★ 867
<b>kint-php/kint</b> Kint - debugging tool for PHP developers	PHP	↓ 92 907 ★ 1 965

**Package type**

antaresproject-component	1
apitool	1
application	3
backbee-bundle	1
behat-extension	4
bitrix-module	1
bundle	2
cakephp-plugin	15
codeigniter-third-party	4
component	1
composer-plugin	1
contao-module	2
contenido-plugin	1
debugger	1
debugging	1

[Show more...](#)

**Tags**



☐ debug 428

☐ debugging 112

☐ laravel 69

☐ php 68

☐ log 63

☐ debugger 56

☐ debugbar 48

☐ var\_dump 46

☐ dump 45

☐ profiler 45

☐ logging 42

☐ nette 42

☐ framework 39



# finding packages

**kint-php/kint**

```
➦ composer require kint-php/kint
```

## Kint - debugging tool for PHP developers



[github.com/kint-php/kint](https://github.com/kint-php/kint)

[Homepage](#)

Source

## Issues

Installs: 93 848

Dependents: 61

Suggesters: 1

Stars: 1 956

Watchers: 123

Forks: 219

Open Issues: 3

## 2.2

2017-09-06 17:46 UTC

requires

- php: >=5.1.2

requires (dev)

- [friendsofphp/php-cs-fixer](#): ^2.0
- [phpunit/phpunit](#): ^4.0
- [symfony/finder](#): ^2.6

suggests

None

provides

None

conflicts

None

replaces

None

© MIT  b091715eadaf6e1a7ef927f3e81d1004611d2aea

dev-master

## 2.2

### 2.1.2

### 2.1.1

## 2.1

2.0

2.0-rc1

2.0-beta2

2.0-beta1

2.0-alpha5

2.0-alpha4

# finding packages

## README

build passing

[-] \$GLOBALS array (12)	→
_GET => array (0)	↗
_POST => array (0)	↗
_COOKIE => array (0)	↗
_FILES => array (0)	↗
_ENV => array (0)	↗
_REQUEST => array (0)	↗
[+] _SERVER => array (31)	→ ↗
⊗ GLOBALS => array (12) Recursion	→ ↗

### What am I looking at?

At first glance Kint is just a pretty replacement for `var_dump()`, `print_r()` and `debug_backtrace()`.

However, it's much, *much* more than that. Even the excellent `xdebug` `var_dump` improvements don't come close - you will eventually wonder how you developed without it.

Just to list some of the most useful features:

### Installation

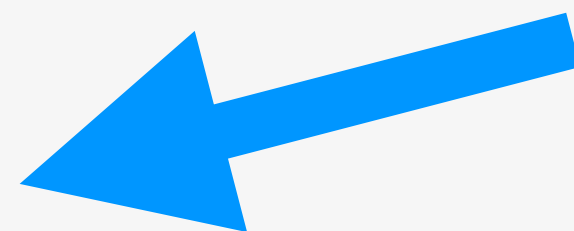
One of the main goals of Kint is to be **zero setup**.

Download the file and simply

```
<?php
require 'kint.php';
```

Or, if you use Composer:

```
composer require kint-php/kint
```



### Usage

# PHP

basic